

1/40

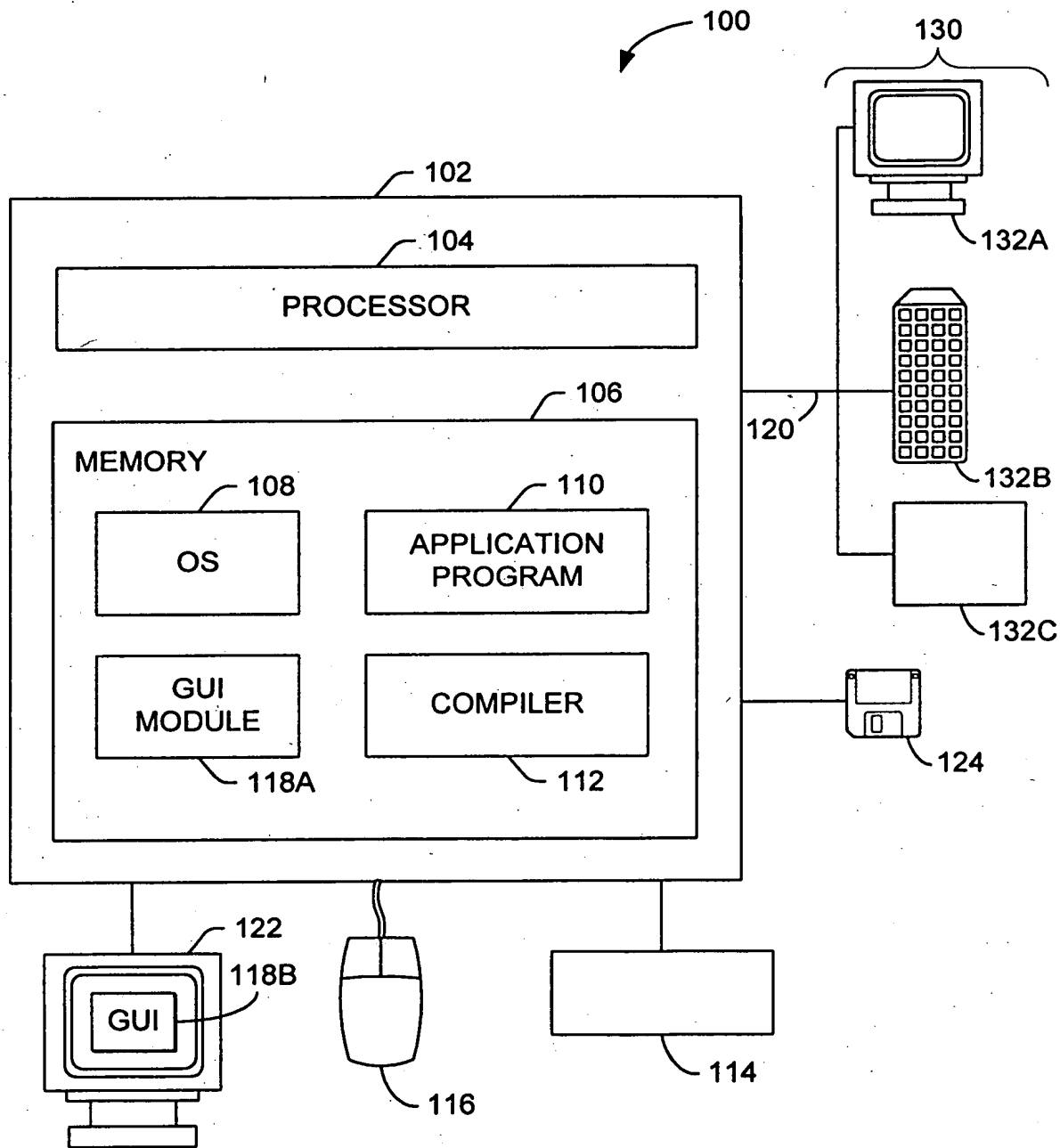


FIG. 1

2/40

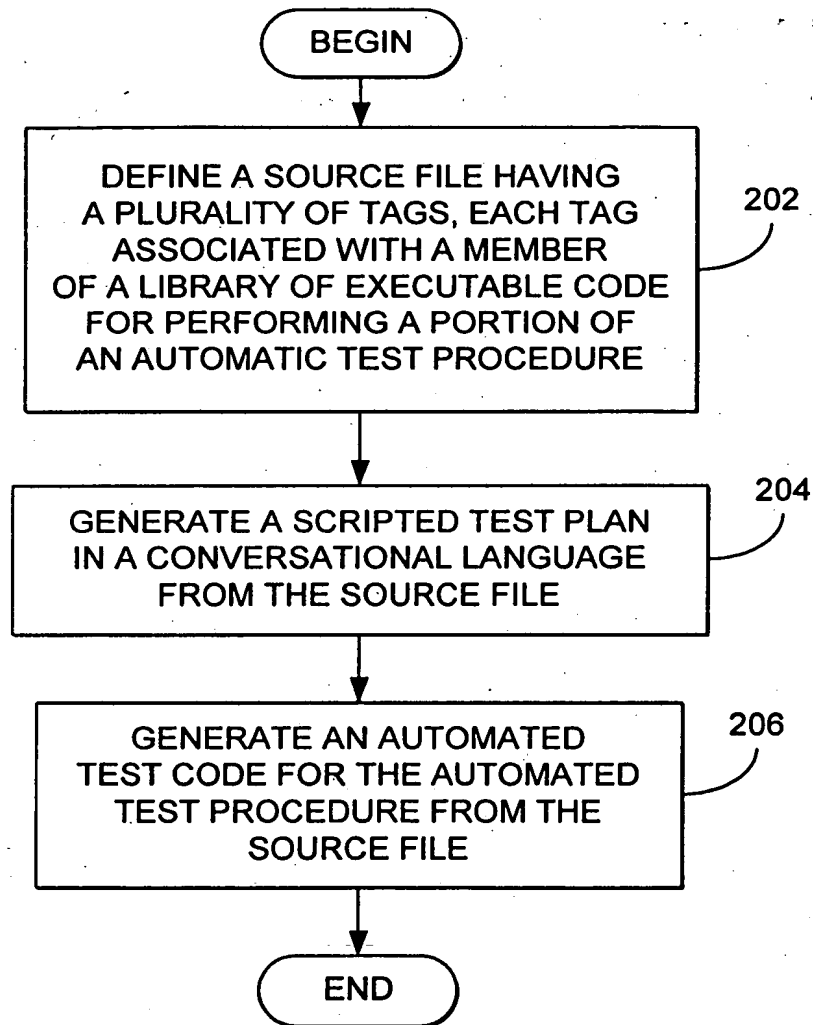


FIG. 2

3/40

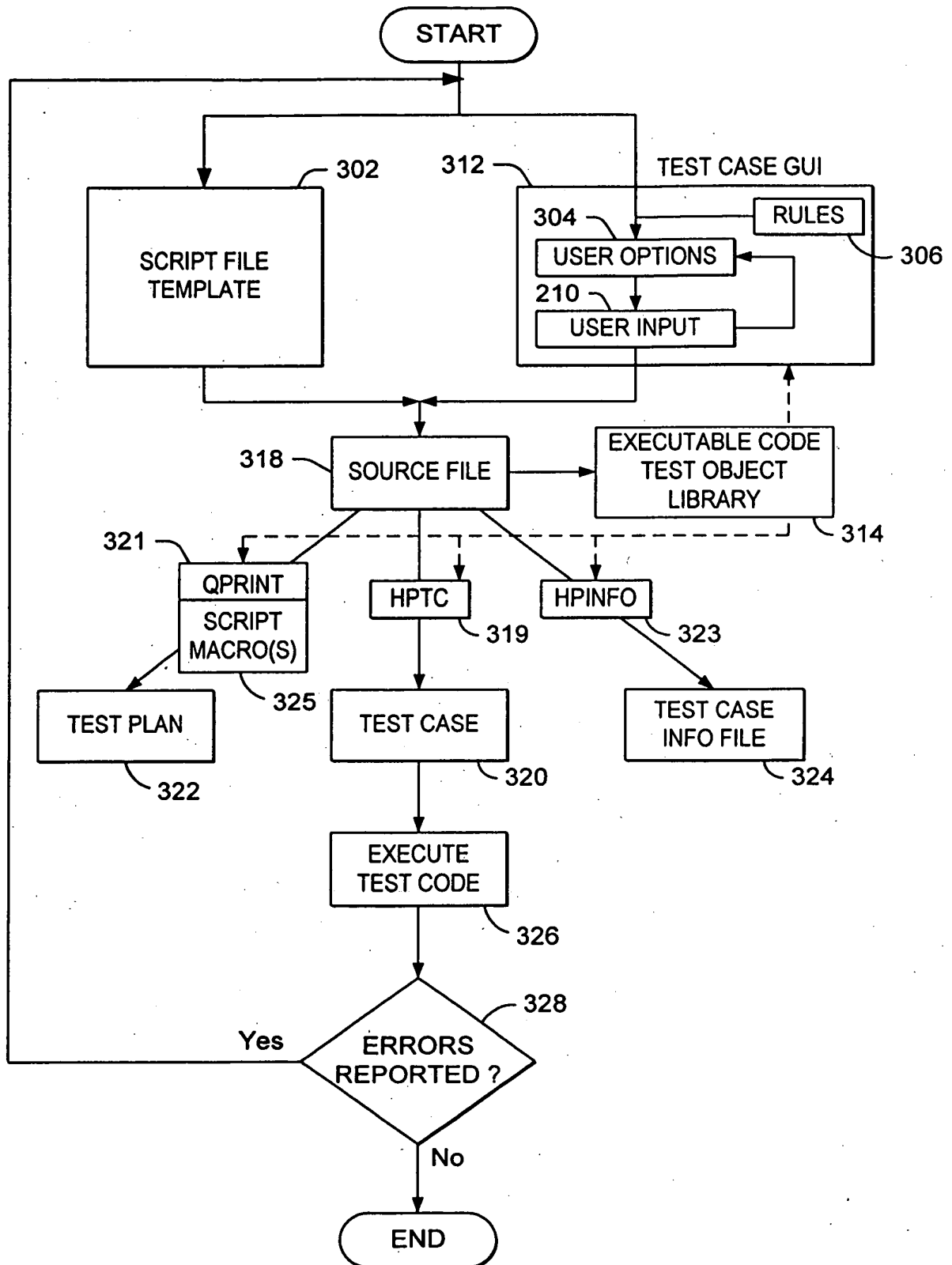


FIG. 3

4/40

302

:H3.SMQA0 1 -9 1 IMS/CQS, non-response mode trans
:hppartc tc= 'SMQA0 1 -9'

402 —:h4.Objectives
404 —:p.The purpose of the test is entered here.
406 —:h4.Scenario
408 —:ol compact.
410 —:li.The test scenario is entered here
412 —:eol.
414 —:h4.Procedure
416 —:ol compact.
418 —:li.Insert IT2 procedures here
420 —:eol.
422 —:h4.Verification
424 —:ol compact.
426 —:li.Testcase is self-verifying.
428 —:eol.
430 —:h4.System Configuration
432 —:ol compact.
434 —:li.This test case uses configuration____.
436 —:eol.
438 —:h4.Parts used by Test Case
440 —:insert parts used by Test Case
442 —:hpauthor aname= Insert author name

FIG. 4

5/40

500

:H3.SMQA0 1 -9 1 IMS/CQS, non-response mode trans

:hppartc tc= 'SMQA0 1 -9'

:h4.Objectives

502 { :p. The purpose of the test is to validate that transactions can be processed on the Shared Message Queue. This test case will queue up non-response mode transactions on the queue and process them.

:h4.Scenario

:ol compact.

504 { :li. Start up a 1-way SYSPLEX with 1 Coupling Facility

:li. Initialize the RECONS and Load the DA Data Base share level 3

:li. Start IRLM 2.1

:li. Cold start 1 IMS/CQS.

:li. Submit 1000 non-response mode transactions

:li. Start application program

:li. Submit another 500 non-response mode transactions

:li. Shutdown IMS after work is processed

:eol.

:h4.Procedure

:ol compact.

:li. Insert IT2 procedures here

:eol.

:h4.Verification

:ol compact.

:li. Testcase is self-verifying.

:eol.

:h4.System Configuration

:ol compact.

:li. This test case uses configuration 30.

:eol.

:h4.Parts used by Test Case

:hppartp

:hpauthor aname= 'Tom Pavela'

FIG. 5

095504-0360

6/40

:H3.SMQA0 1 -9 1 IMS/CQS, non-response mode trans

:hpartc tc= 'SMQA0 1 -9'

:h4.Objectives

:p.The purpose of this test is to validate that transactions can be processed on the Shared Message Queue. This test case will queue up non-response mode transactions on the queue and process them.

:h4.Scenario

:ol compact.

:li.Start up a 1-way SYSPLEX with 1 Coupling Facility

:li.Initialize the RECONS and Load the DJK Data Base share level 3

:li.Start IRLM 2.1

:li.Cold start 1 IMS/CQS.

:li.Submit 1000 non-response mode transactions

:li.Start application program

:li.Submit another 500 non-response mode transactions

:li.Shutdown IMS after work is processed

:eol.

:h4.Procedure

:ol compact.

602A ~:HPENTRY CONFIG=30 ~ 602B

604A ~:HPLOAD DB='DJK' SHRL=3 ~ 604B

606A ~:HPSRLM2 ON=ALL ~ 606B

608A ~:IMSSTART ON=ALL DB='DJK' RE=NRE ~ 608B

CFNAMES1 =CFNAMES,CFIRLM=LT01,CFVSAM=,CFOSAM=OSAMSESXI'

610A ~:TMSCONTI ON=ALL NTRANS=1000 ~ 610B

612A ~:TMSCONTI ON=ALL NTRANS=500 ~ 612B

614A ~:IMSSTOP ON=ALL ~ 614B

616A ~:HPEXIT ~ 616B

:eol.

:h4.Verification

:ol compact.

:li.Testcase is self-verifying.

:eol.

:h4.System Configuration

:ul compact.

:li.This test case uses configuraton 30.

:eul.

:h4.Parts used by Test Case

:hpartp

:hpauthor aname= 'Tom Pavela'

FIG. 6

TOP SECRET 40955660

7/40

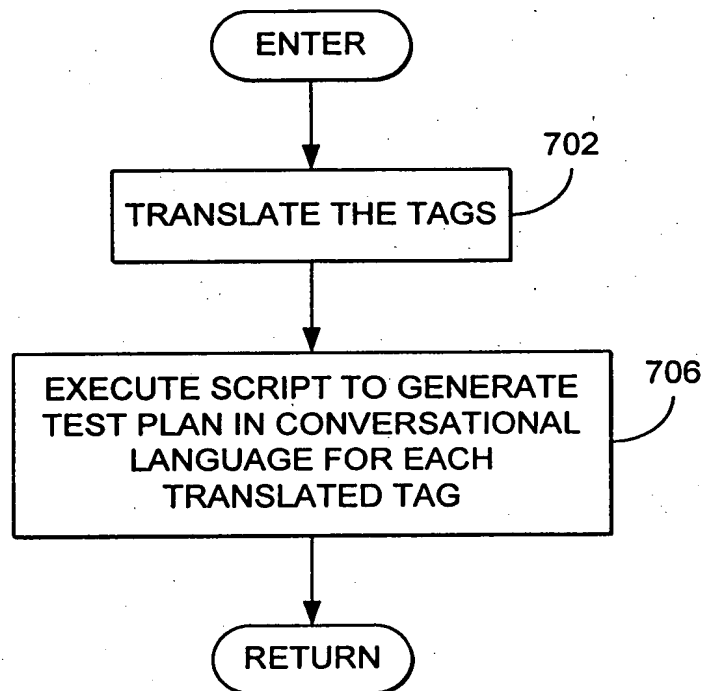


FIG. 7

8/40

SMQA0 1 -9 1 IMS/CQS, NON-RESPONSE MODE TRANS

Objectives

The purpose of this test is to validate that transactions can be processed on the Shared Message Queue.

This test case will queue up non-response mode transactions on the queue and process them.

Scenario

1. Start up a 1-way SYSPLEX with 1 Coupling Facility
2. Initialize the RECONS and Load the DJK Data Base share level 3
3. Start IRLM 2.1
4. Cold start 1 IMS/CQS.
5. Submit 1000 non-response mode transactions
6. Start application program
7. Submit another 500 non-response mode transactions
8. Shutdown IMS after work is processed

Procedure

1. Call Hpcs_entry using configuration 30 and ARM= NO and ARCDEFLT= YES and RECVTAM= YES
2. Call Hpcs_load_databases which will:
 - a. Define the shared RECON data sets
 - b. Run the load database job(s) to load database(s) DJK and register the data bases as share level -3
3. Call Hpcs_Start_IRLMs_21 which will:
 - a. Start IRLM 2.1 on all CECs with a lock structure of LT01
4. Call Start_IMS_on_all_systems which will:
 - a. Run HPC\$SPEC MODEL to update the VSPEC member on all CECs with (CFNAMES,CFIRLM=LT01,CFVSAM=,CFOSAM=OSAMSESXI)
 - b. Run IMS%CSA% MVSPROC to bring up IMS TM/DB region on all CECs with CQS using VCATSHR.
 - c. After DFS810A message is displayed, issue "/NRE CHKPT 0 FORMAT ALL." Wait for cold start to complete.
 - 1) Issue IMS command "/STOP DB DBHDOJ01"
 - 2) Issue IMS command "/STOP DB DBHDOK01"
 - 3) Issue IMS command "/START DB DBHDOJ01 ACCESS=UP"
 - 4) Issue IMS command "/START DB DBHDOK01 ACCESS=UP"

FIG. 8A

T06T60"40355660

9/40

5. Call Start_Transaction_Scenario_1 which will:
 - a. Submit 1000 non-response mode transactions (HPCSTCL1) on all CECs
 - b. Issue the IMS / START PROGRAM HPC\$M\$00 command on all CECs
 - c. Wait for all Scenario 1 transactions to be processed, then verify the transaction counter is correct.
6. Call Start_Transaction_Scenario_1 which will:
 - a. Submit 500 non-response mode transactions (HPCSTCL1) on all CECs
 - b. Issue the IMS / START PROGRAM HPC\$M\$00 command on all CECs
 - c. Wait for all Scenario 1 transactions to be processed, then verify the transaction counter is correct.
7. Call Stop_all_IMSs which will:
 - a. Issue a "/CHE FREEZE" to bring down the IMS control region on all CECs
 - b. When IMS control region on all CECs completes, verify all condition codes to be zero.
 - c. In Shared Queues configurations when CQS region on all CECs completes, verify all condition codes to be zero.
8. Call Hpcs_Exit routine

Verification

1. Testcase is self-verifying.

System Configuration

- o This test case uses configuration 30.

Parts used by Test Case

CFCPLOAD PROCEDURE
 RCN%CSA% PROCEDURE
 HPC\$L05 MODEL
 LOADDJK PROCEDURE
 IRLME2N PROCEDURE
 HPC\$SPEC MODEL
 IMS%CSA% PROCEDURE
 SMQ\$C19X MVSPROC
 SMQ\$BMP JCL
 HPC\$TPNS MODEL
 HPC\$MPP MODEL
 HPC\$JOB EXEC

Author: Tom Pavela

FIG. 8B

T06T60"40B5560

10/40

```

/*****
/*SMQA0 1 -9 1 IMS/CQS, non-response mode trans
/*****
/*
/* Objectives
/*
/* The purpose of this test is to validate that transactions can be
/* processed on the Shared Message Queue. This test case will queue
/* up non-response mode transactions on the queue and process them.
/*
/* Scenario
/*
/* Start up a 1-way SYSPLEX with 1 Coupling Facility
/*
/* Initialize the RECONS and Load the DJK Data Base share level
/* 3
/*
/* Start IRLM 2.1
/*
/* Cold start 1 IMS/CQS.
/*
/* Submit 1000 non-response mode transactions
/*
/* Start application program
/*
/* Submit another 500 non-response mode transactions
/*
/* Shutdown IMS after work is processed
/*

```

FIG. 9A

0005304-091901

11/40

```

/***** */
/* This TC requires that an EC machine be ipled and executing in a */
/* Parallel SYSPLEX Environment (with a Coupling Facility) */
/*===== */
/* SECURITY : IBM INTERNAL USE ONLY */
/*===== */
/* TESTCASE NAME : "SMQA01-9" */
/*===== */
/* SOURCE FILE : "SMQA01-9 SCRIPT A1" */
/*===== */
/* LINE ITEM : 1 IMS/CQS, non-response mode trans */
/*===== */
SESSION=SESSION
GLOBAL SetVars MoreHold DoReply DoWait TimeOut SwitchEC ATIRUN
GLOBAL MVSPROC ATIVER TransVer MVSCmd CPCmd GoCP LeaveCP GetPRTAll
GLOBAL ResetPorts DialPorts LogLine
GLOBAL DoWaitSwap ATISwap MVSCmdSwap
/***** */
/* Hpcs subroutine library */
/*===== */
GLOBAL CONFIGURATION /*: determines #ECs & #CFs & struct location */
GLOBAL DATABASES /*: determines databases to load and access */
GLOBAL CFNAMES1 /*: CFNAMES card #1 used by HPC$VSPEC */
GLOBAL CFNAMES2 /*: CFNAMES card #2 used by HPC$VSPEC */
GLOBAL OPTIONS /*: IRLM 2.1 options (start_a_Irlm only) */
GLOBAL DBDLIST /*: dbdlist at hpcs_entry */
GLOBAL ACBLIB /*: acblib at hpcs_entry, psb will be gened to */
GLOBAL HPCLIST /*: psblist at hpcs_entry */
GLOBAL HPCSTRCE /*: TRACE value while in HPC$SUB */
/*-----Scenario variables----- */

```

FIG. 9B

12/40

GLOBAL HPCSLOG	/*: LOG causes Scenario logging to OLDS	*/
GLOBAL HPCSTRAN	/*: #trans to use in Scenario 1-350	*/
	/* Scenarios 4, 5 <176 else <351	*/
GLOBAL HPCSMPPS	/* The number of Mpps to be used by	*/
	/* database type, 1, 2 or 3 (def=3)	*/
GLOBAL HPCSVER	/* ='Yes' verify environment, ='No', goto	*/
	/* check all messages processed loop	*/
GLOBAL HYPER	/* Yes--use VSAM Hyper space	*/
GLOBAL SHARER	/* Yes--start 2nd IMS	*/
GLOBAL MODEL	/* Mvscmd model proc	*/
GLOBAL NUMPARTS	/* Number of Partitions	*/
GLOBAL RESLIB	/* IMS reslib	*/
GLOBAL PARM1	/* IMS parm1	*/
GLOBAL PARM2	/* IMS parm2	*/
GLOBAL RESTART_VTAM	/* restart_vtam=yes/no for recycled vtam in entry	*/
GLOBAL VSPEC	/* IMS VSPEC	*/
GLOBAL PROCNAME	/* IMS PROCNAME	*/
GLOBAL CEC1_RESLIB	/* CEC1 RESLIB	*/
GLOBAL CEC2_RESLIB	/* CEC2 RESLIB	*/
GLOBAL CEC3_RESLIB	/* CEC3 RESLIB	*/
GLOBAL Scenario2_log	/* TMSCNTX Scenario2 _Log	*/
GLOBAL Scenario3_log	/* TMSCNTX Scenario3 _Log	*/
GLOBAL Scenario4_log	/* TMSCNTX Scenario4 _Log	*/
GLOBAL Scenario5_log	/* TMSCNTX Scenario5 _Log	*/
GLOBAL Scenario6_log	/* TMSCNTX Scenario6 _Log	*/
GLOBAL Scenario7_log	/* TMSCNTX Scenario7 _Log	*/
GLOBAL Scenario8_log	/* TMSCNTX Scenario8 _Log	*/
GLOBAL Scenario9_log	/* TMSCNTX Scenario9 _Log	*/
GLOBAL ScenarioA_log	/* TMSCNTX ScenarioA _Log	*/
GLOBAL ScenarioB_log	/* TMSCNTX ScenarioB _Log	*/
GLOBAL ScenarioC_log	/* TMSCNTX ScenarioC _Log	*/
GLOBAL ScenarioD_log	/* TMSCNTX ScenarioD _Log	*/
GLOBAL ScenarioE_log	/* TMSCNTX ScenarioE _Log	*/

FIG. 9C

13/40

```

GLOBAL ScenarioF_log      /* TMSCNTX ScenarioF_Log      */
GLOBAL ScenarioG_log      /* TMSCNTX ScenarioG_Log      */
GLOBAL ScenarioH_log      /* TMSCNTX ScenarioH_Log      */
GLOBAL ScenarioI_log      /* TMSCNTX ScenarioI_Log      */
GLOBAL ScenarioJ_log      /* TMSCNTX ScenarioJ_Log      */
GLOBAL ScenarioK_log      /* TMSCNTX ScenarioK_Log      */
GLOBAL ScenarioL_log      /* TMSCNTX ScenarioL_Log      */
GLOBAL CQSWTOR1           /* CQSWTOR1                    */
GLOBAL ARCDEFLT           /* Archive member default     */
GLOBAL NumofTerm_to_Use   /* Num of Terminal to use for Scenario 1-10 */
GLOBAL ARM                /* ARM policy                  */
GLOBAL ShareDB            /* Share DB YES-Global No-Local */
GLOBAL IMSLOCAL           /* Local IMS?                  */
GLOBAL RSRMBR            /* RSRMBR RSR Member          */
GLOBAL DELSLDS            /* DELSLDS Delete SLDS        */
GLOBAL RLVL               /* RLVL Readiness level       */
GLOBAL HPCSFRCE           /* routine in HPC$CMD to cleanup structures */
/*****
/*-----Called Commands-----
GLOBAL SwitchEC
GLOBAL Hpcs_entry
GLOBAL Hpcs_load_databases
GLOBAL Hpcs_Start_IRLMs_21
GLOBAL Start_IMS_on_all_systems
GLOBAL Start-Tran_Scenario_1
GLOBAL Stop_all_IMSs
GLOBAL Hpcs_exit
GLOBAL Hpcs_clear
GLOBAL Hpcs_clear_all
GLOBAL Hpcs_logit

```

FIG. 9D

14/40

```

/*=====begin test case===== */
/*----->>> EC1 <<<----- */
Call SwitchEC "EC1"
CONFIGURATION=30
RESTART_VTAM="YES"
ARCDEFLT="YES"
ARM="NO"
Call Hpcs_entry " "
/* load the database(s) using sharelevel 3 */
DATABASES=" DJK "
ShareDB="YES"
Call Hpcs_load_databases "3 " 910
Call Hpcs_Start_IRLMs_21 " "
/*===== */
/* Cold start IMS TM_DB region on ALL system(s) */
/* CQS will be started and the default model is SMQ$C19X. */
/* The following IMS parms will be used if they are not set by the */
/* user in IMSPARMS: */
/*     IRLM=Y, VSPEC=HP, IMSID=IMSx */
/*     SHAREDQ=%%x, DC=COx */
/*     note: x is 1,2, or 3 depending on which CEC */
/*     DLINM=HPC%CSA% (if DBDLIST or PSBLIST is specified in HPENTRY) */
/*===== */
CFNAMES1= 'CFNAMES, CFIRLM=LT01, CFVSAM=, CFOSAM=OSAMSESXI'
CFNAMES2="NO"
DATABASES=" DJK "
SHARER="NO"
HYPER="NO"
IMSLOCAL="N"
RESLIB="C"
PROCNAME="DEFAULT"
PARM1=" "
PARM2=" "
VSPEC="DEFAULT"
MODEL="DEFAULT"
Call Start_IMS_on_all_systems
Call Start_Trans_Scenario_1 "LEAVE=NO NTRANS=1000 ON=ALL STARTAPL=ALL"
Call Start_Trans_Scenario_1 "LEAVE=NO NTRANS=500 ON=ALL STARTAPL=ALL"
    Call Stop_all_IMSs " "
Call Hpcs_exit " "
/*=====end test case===== */
EXIT 0
INCLUDE "HPC$SUB"
/*=====HPTC Translation summary===== */
/* Number of lines written = ...176 */
/* Number of +++ errors = .....0 */
/*=====End Translation summary===== */

```

FIG. 9E

0055804 091901

15/40

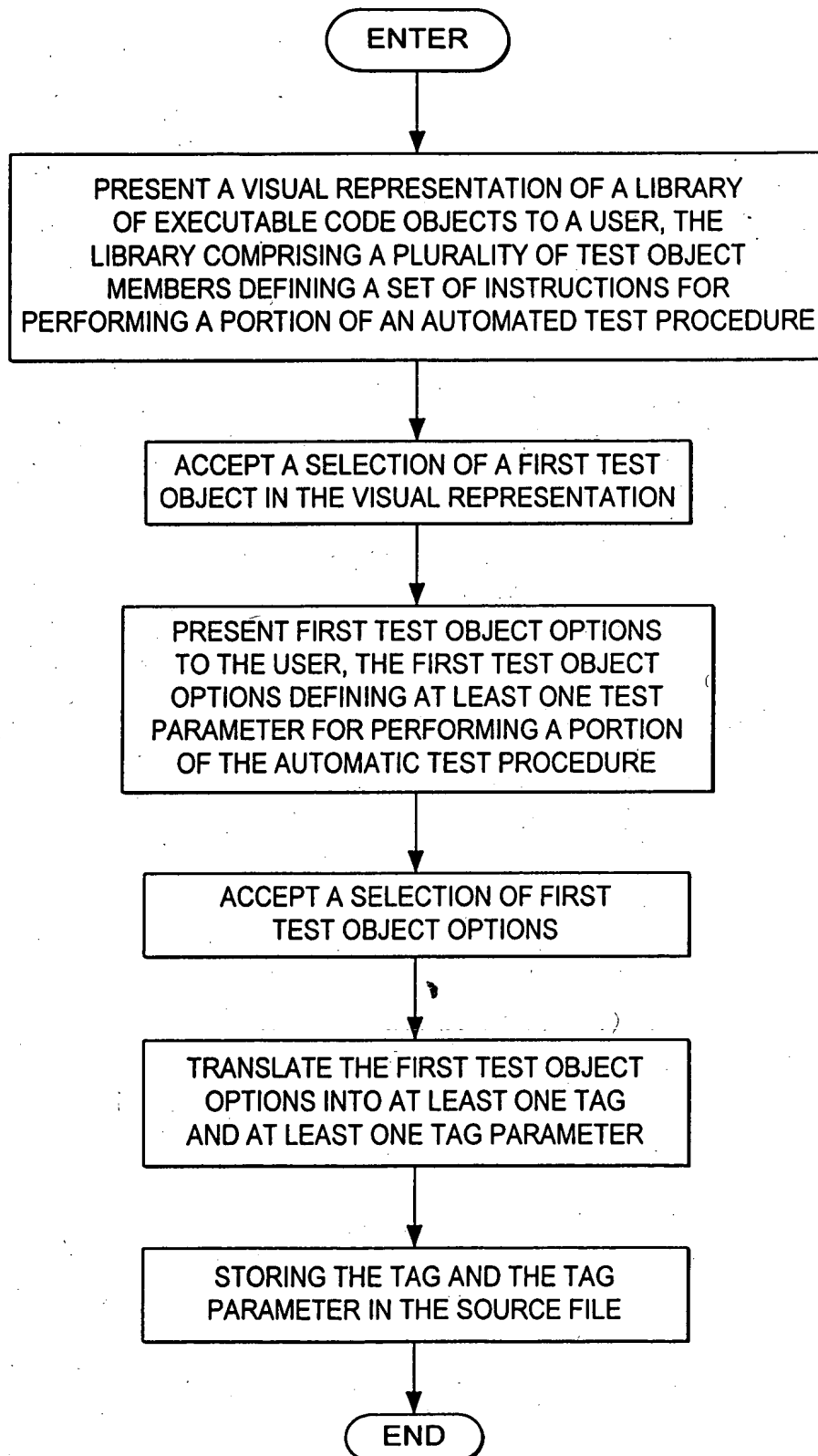


FIG. 10

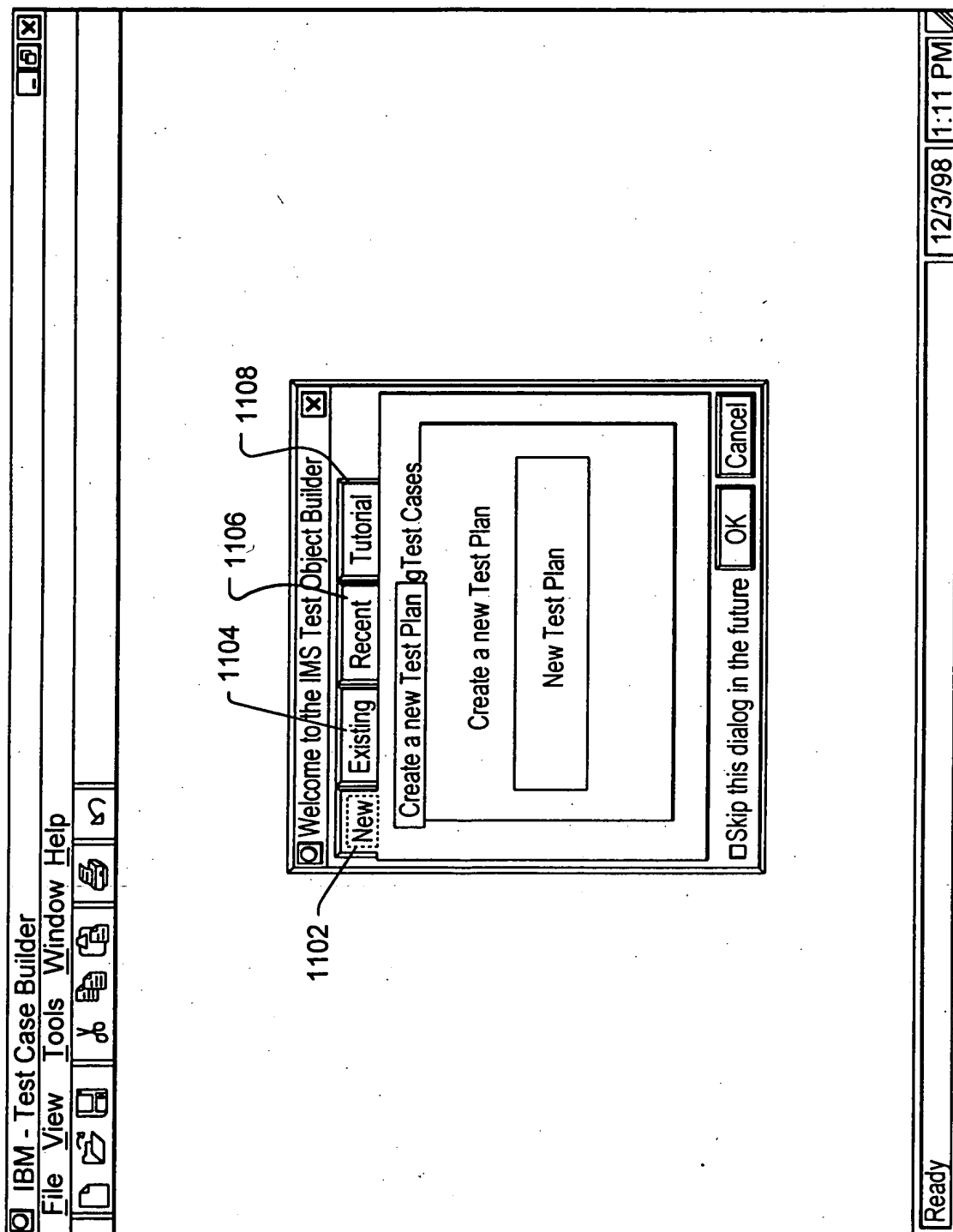


FIG. 11

17/40

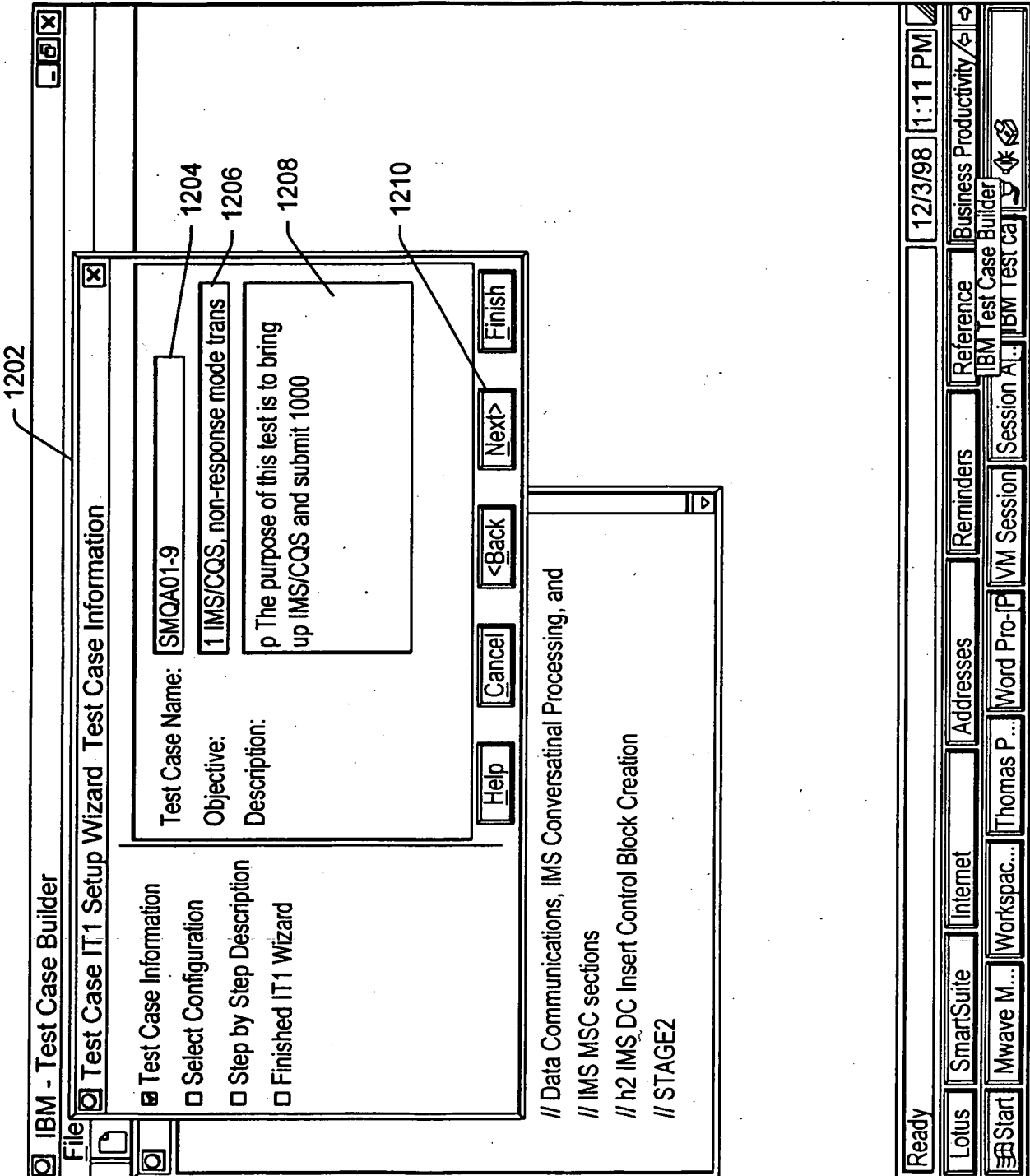


FIG. 12

TOP SECRET 4085960

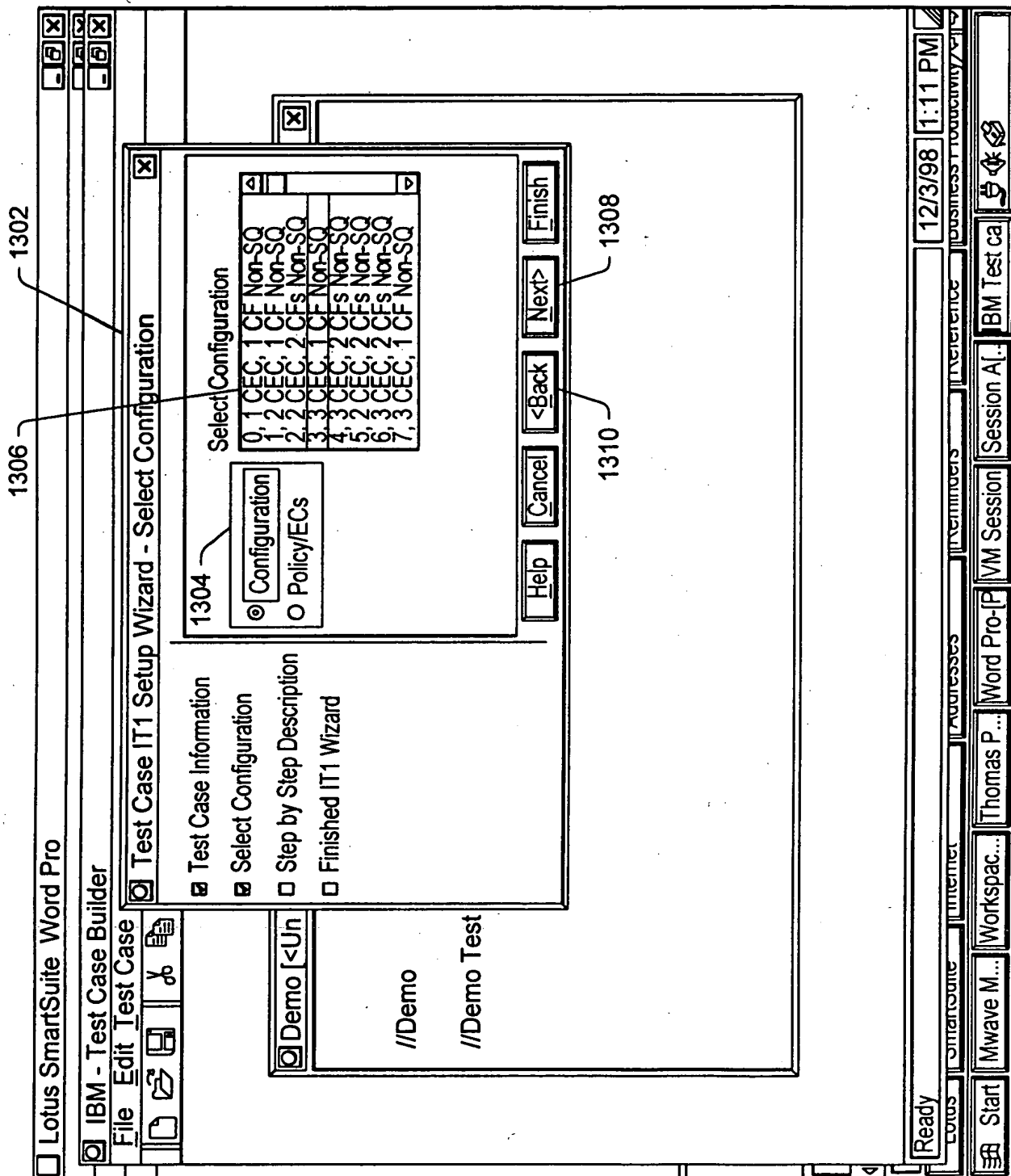


FIG. 13

19/40

1402

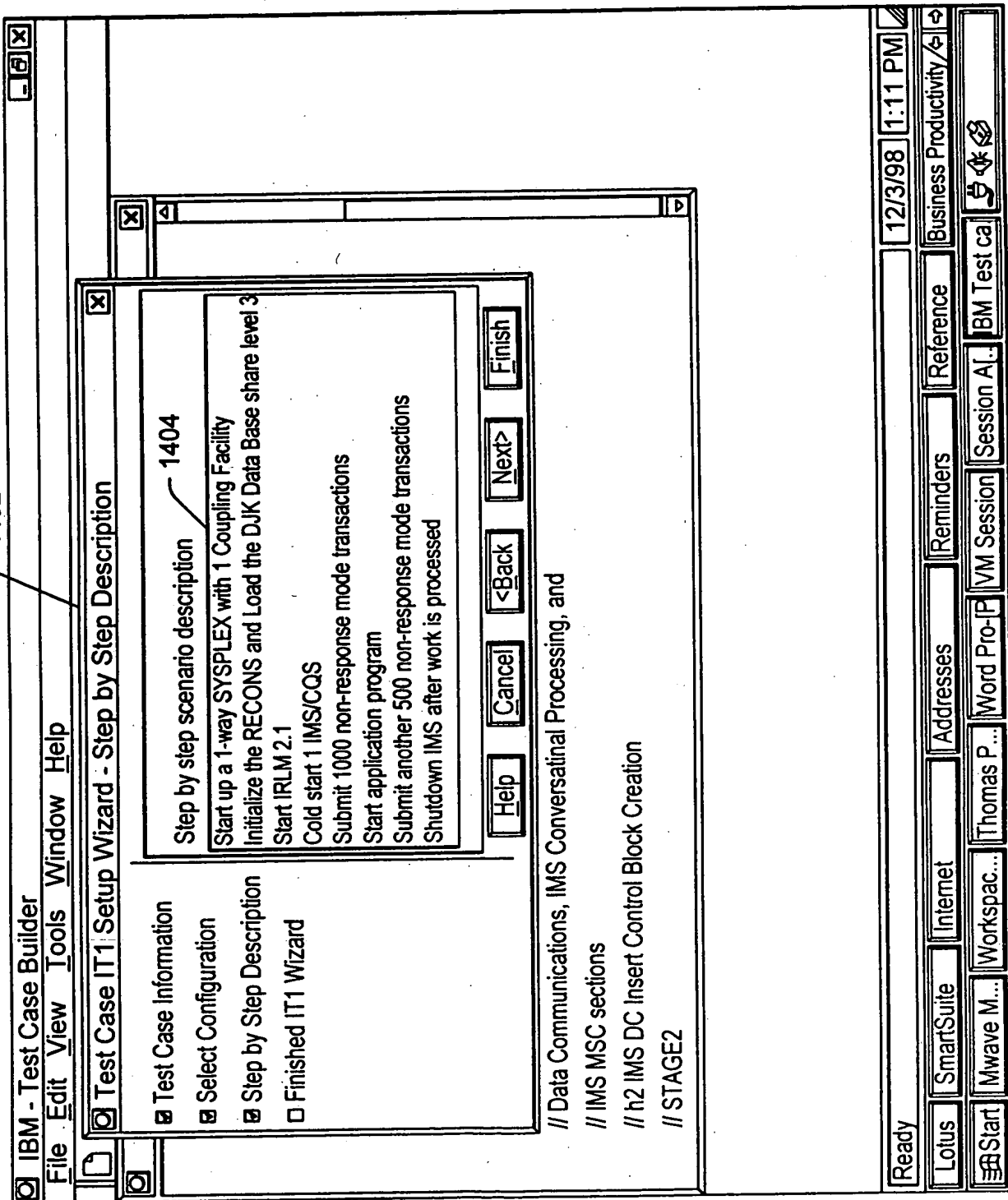


FIG. 14

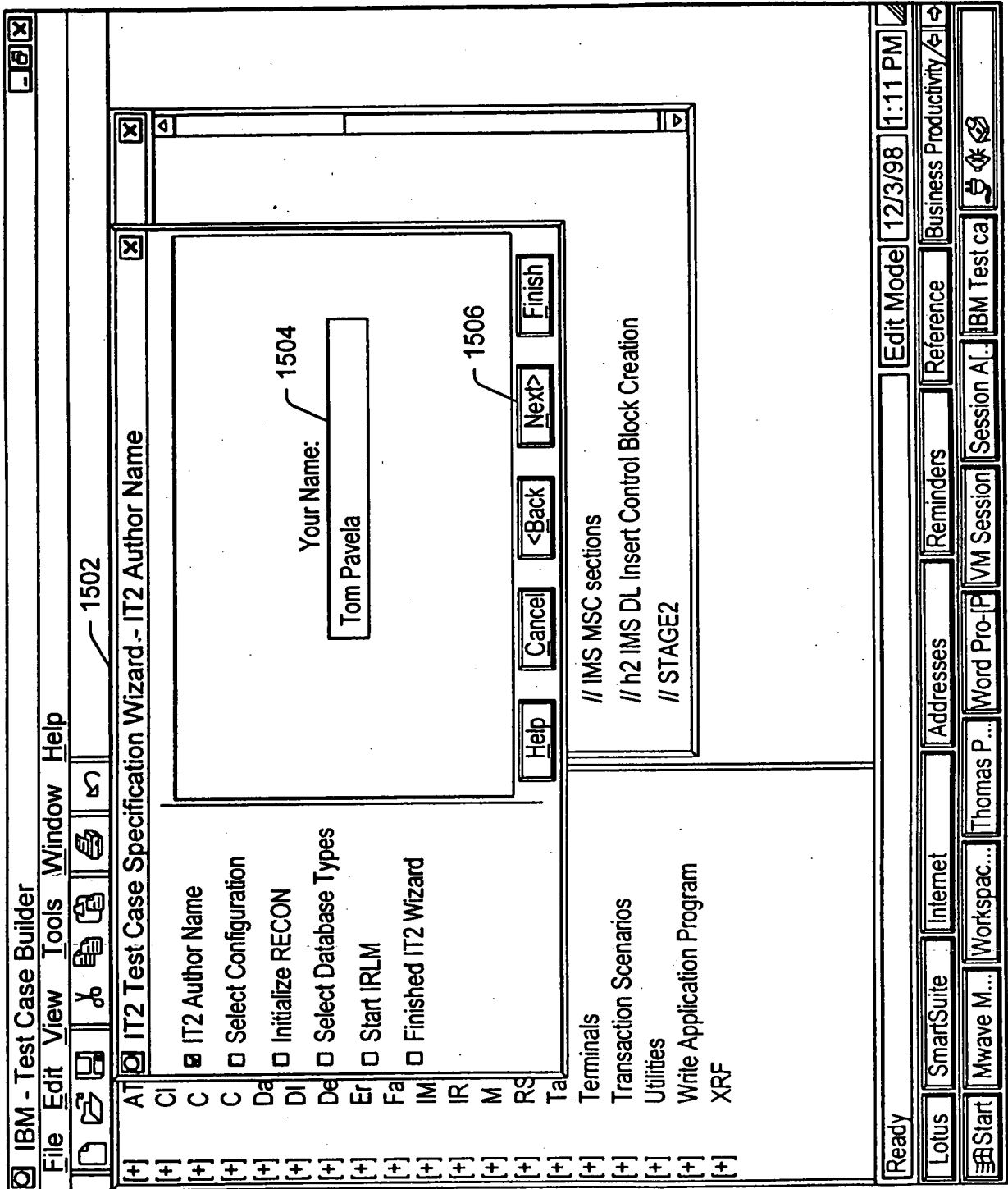


FIG. 15

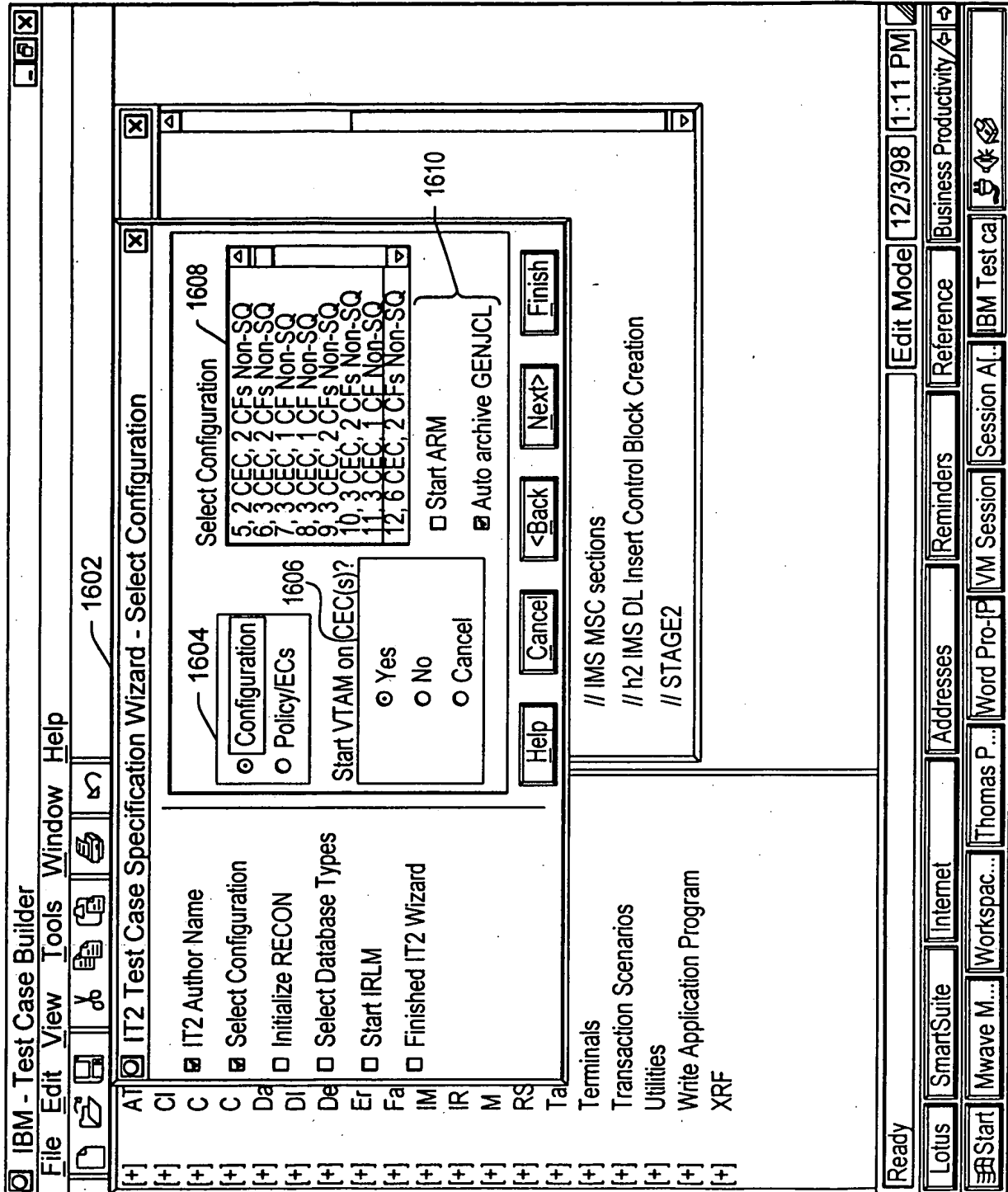


FIG. 16

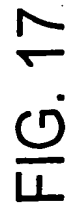


FIG. 17

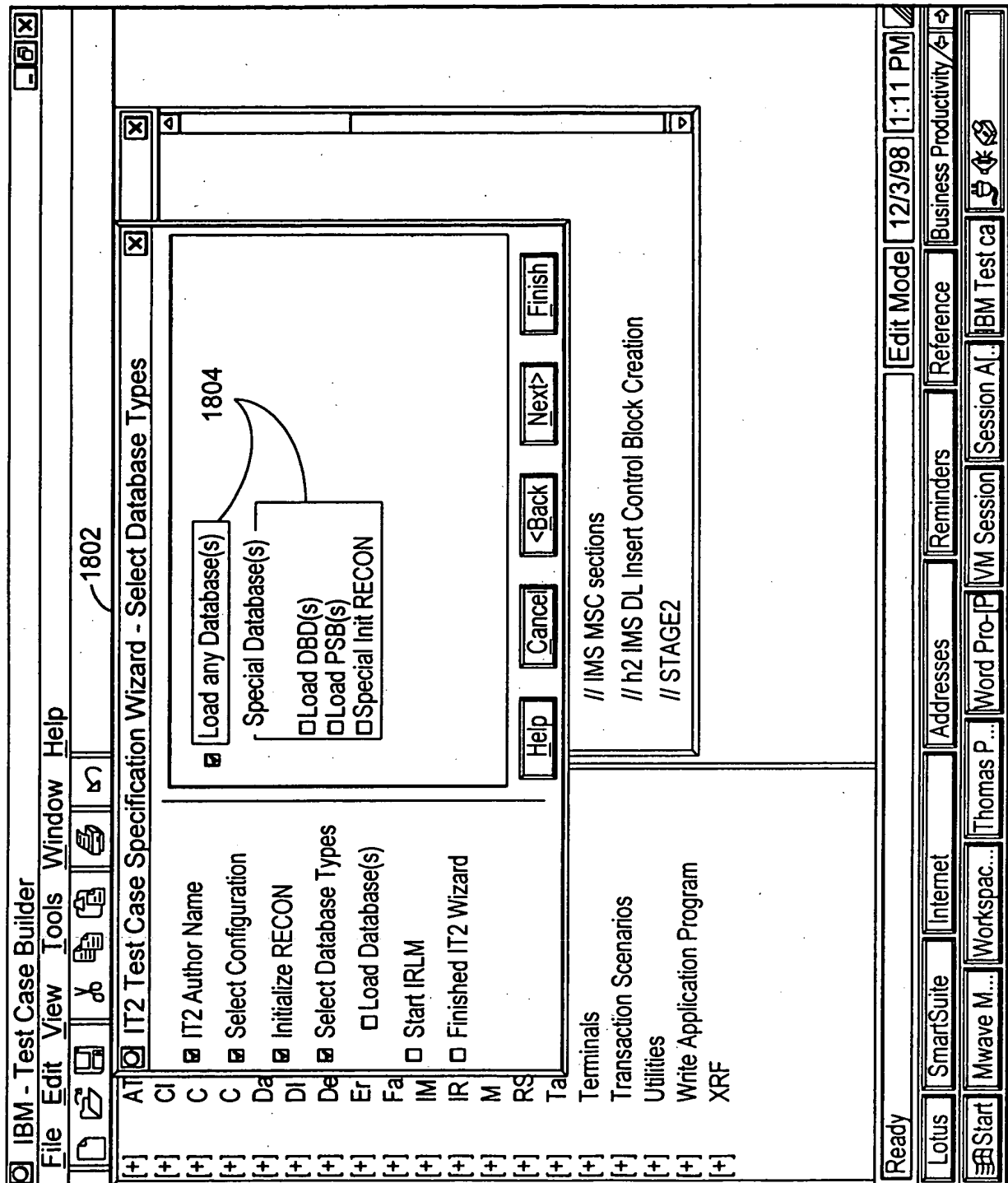


FIG. 18

TOP SECRET 408555

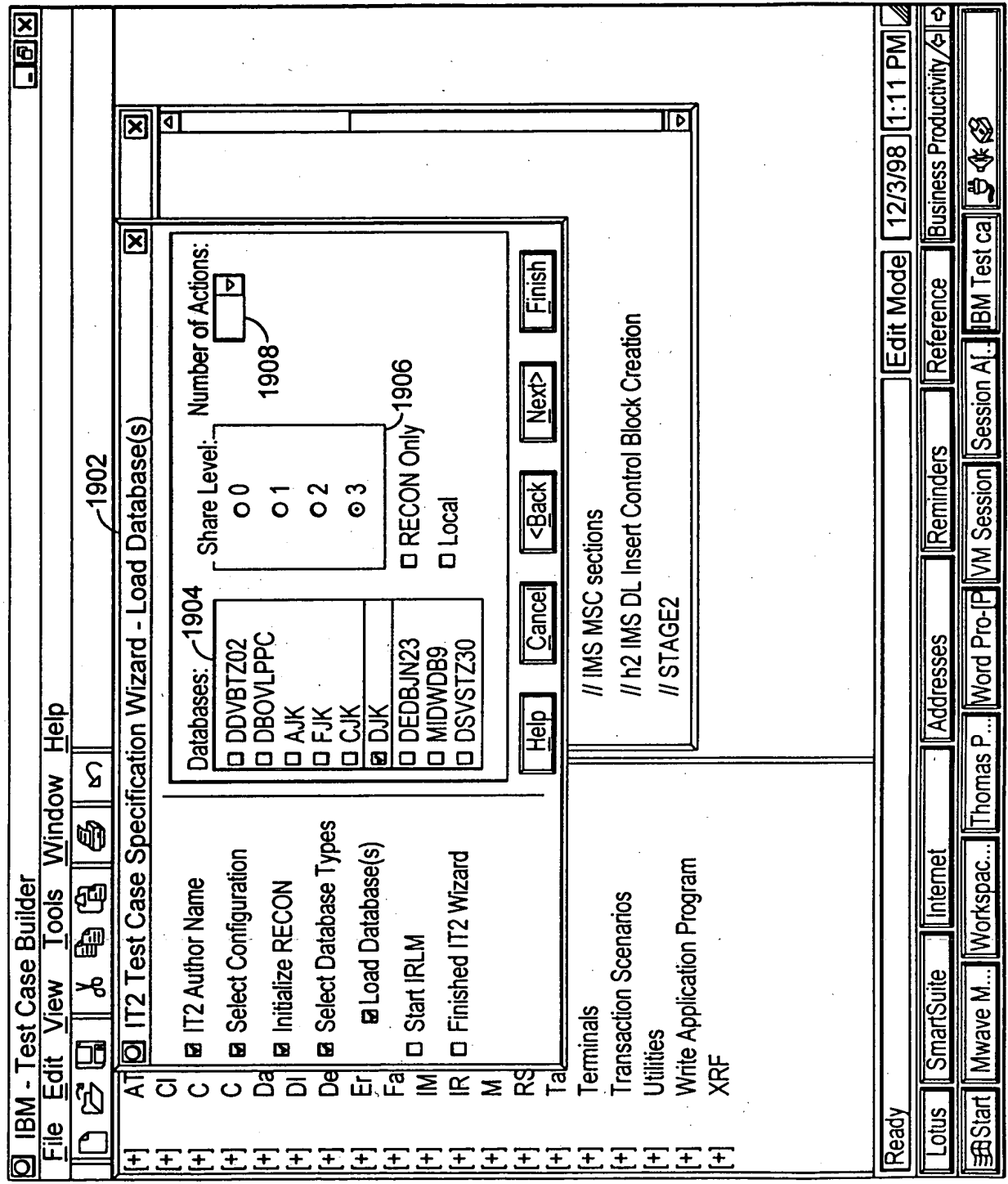


FIG. 19

TOP 405560

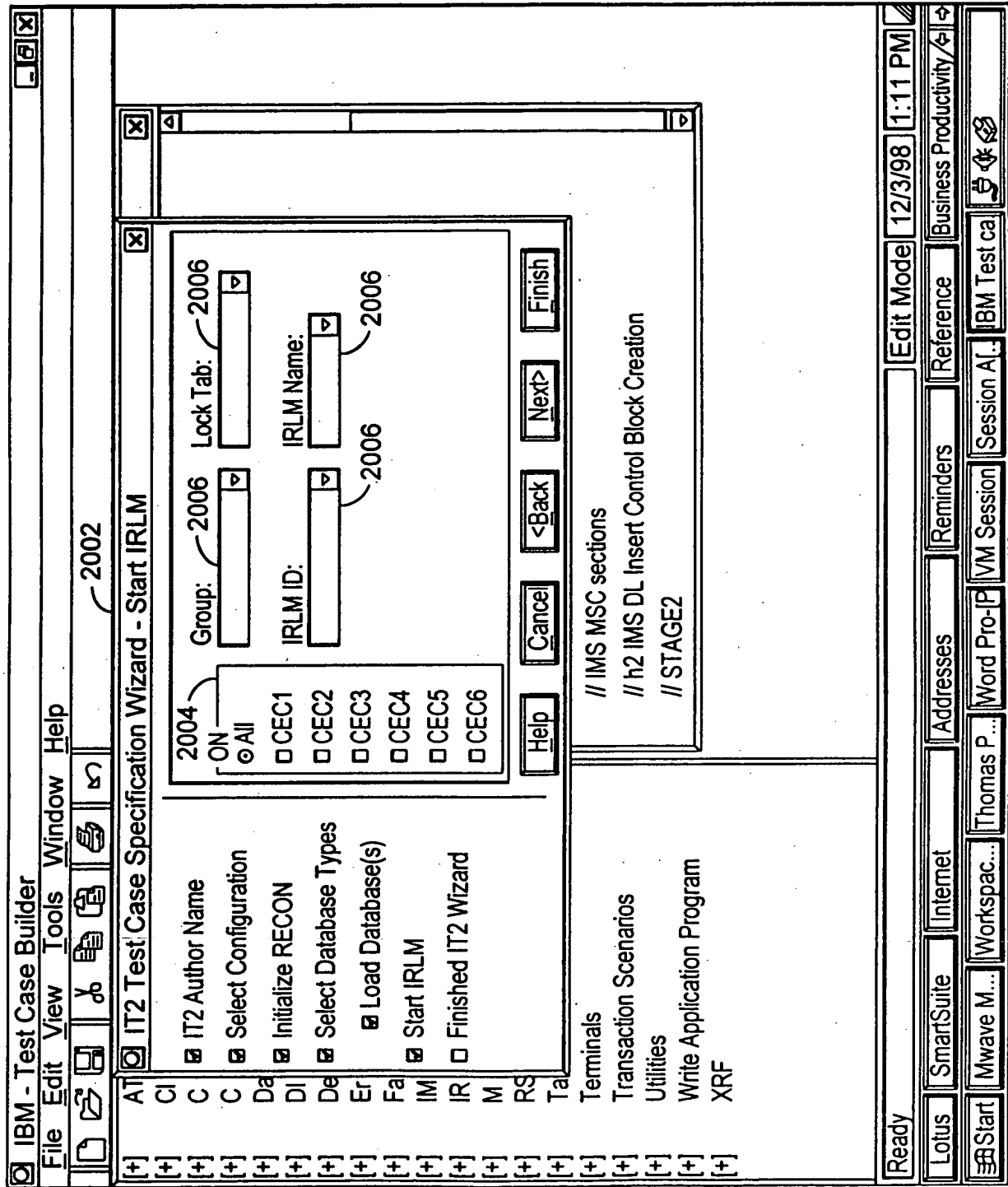


FIG. 20

26/40

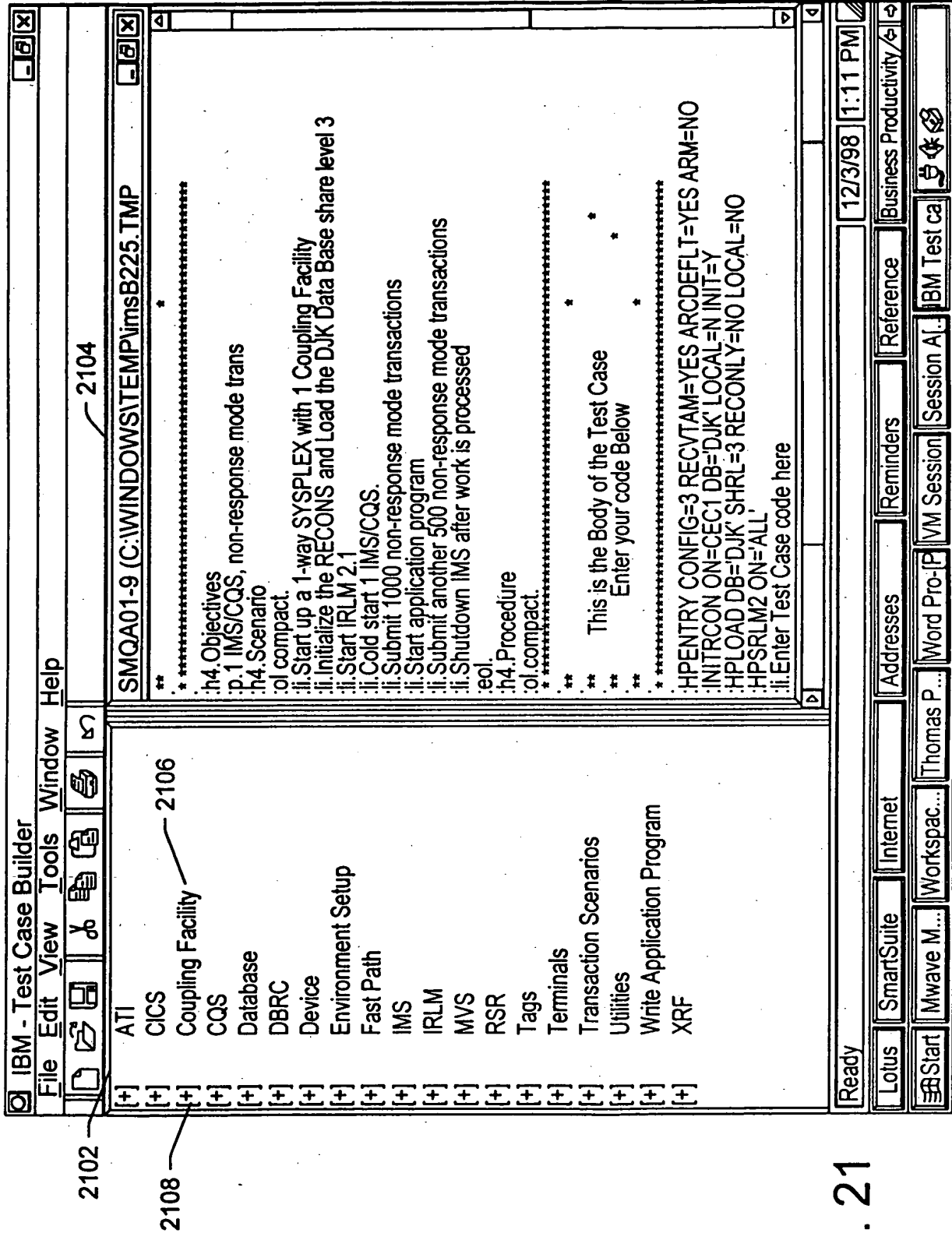


FIG. 21

27/40

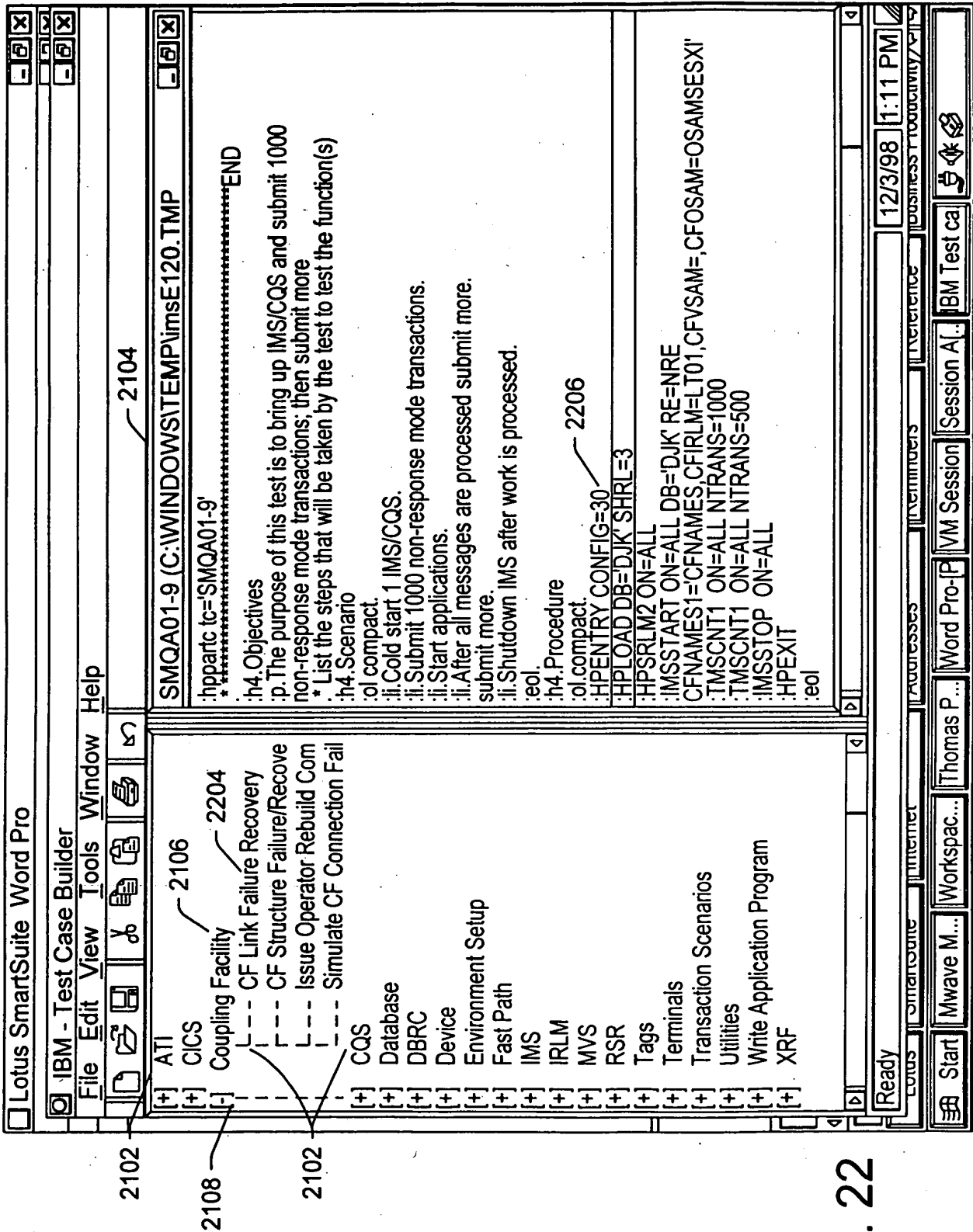


FIG. 22

28/40

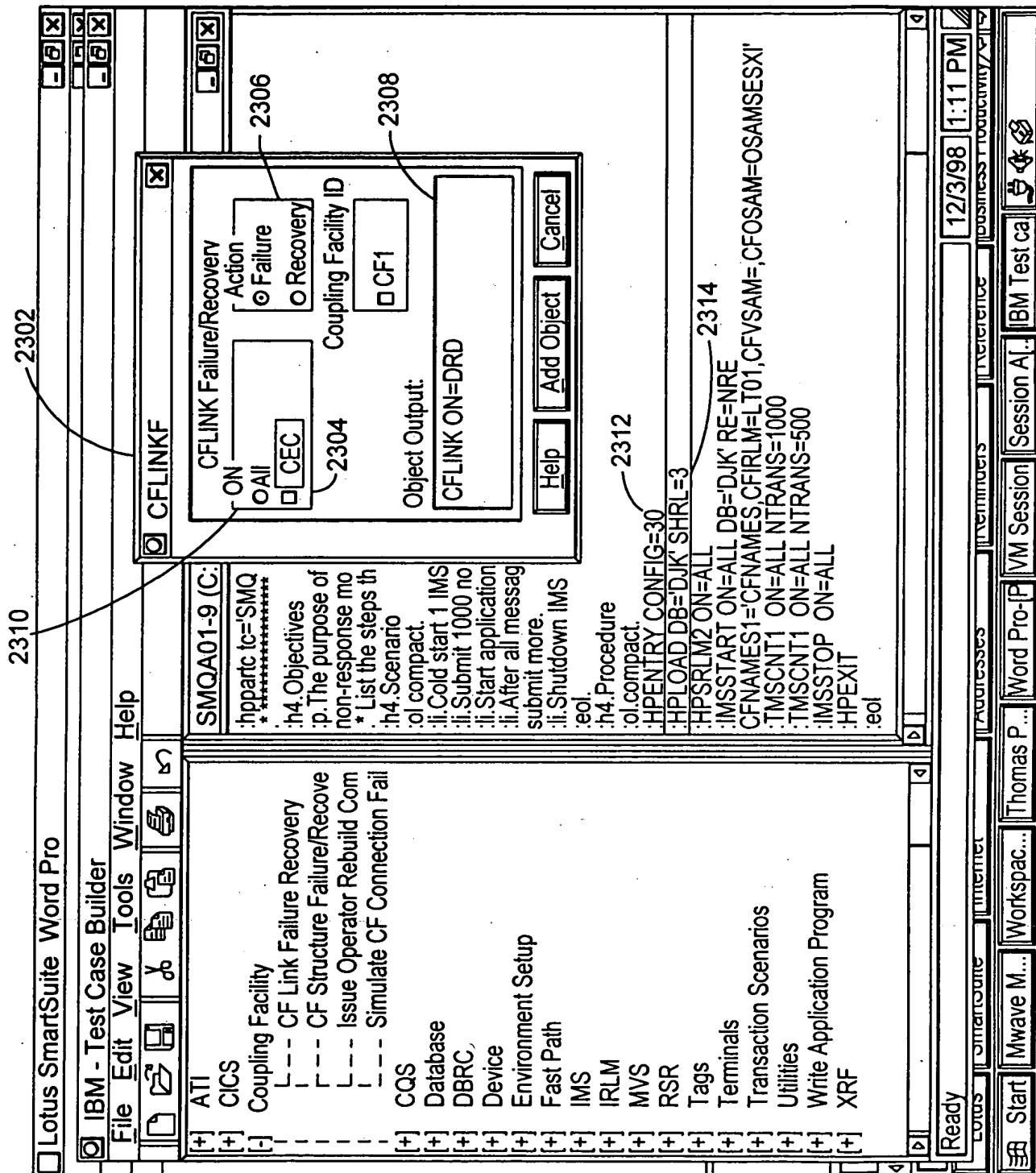


FIG. 23

29/40

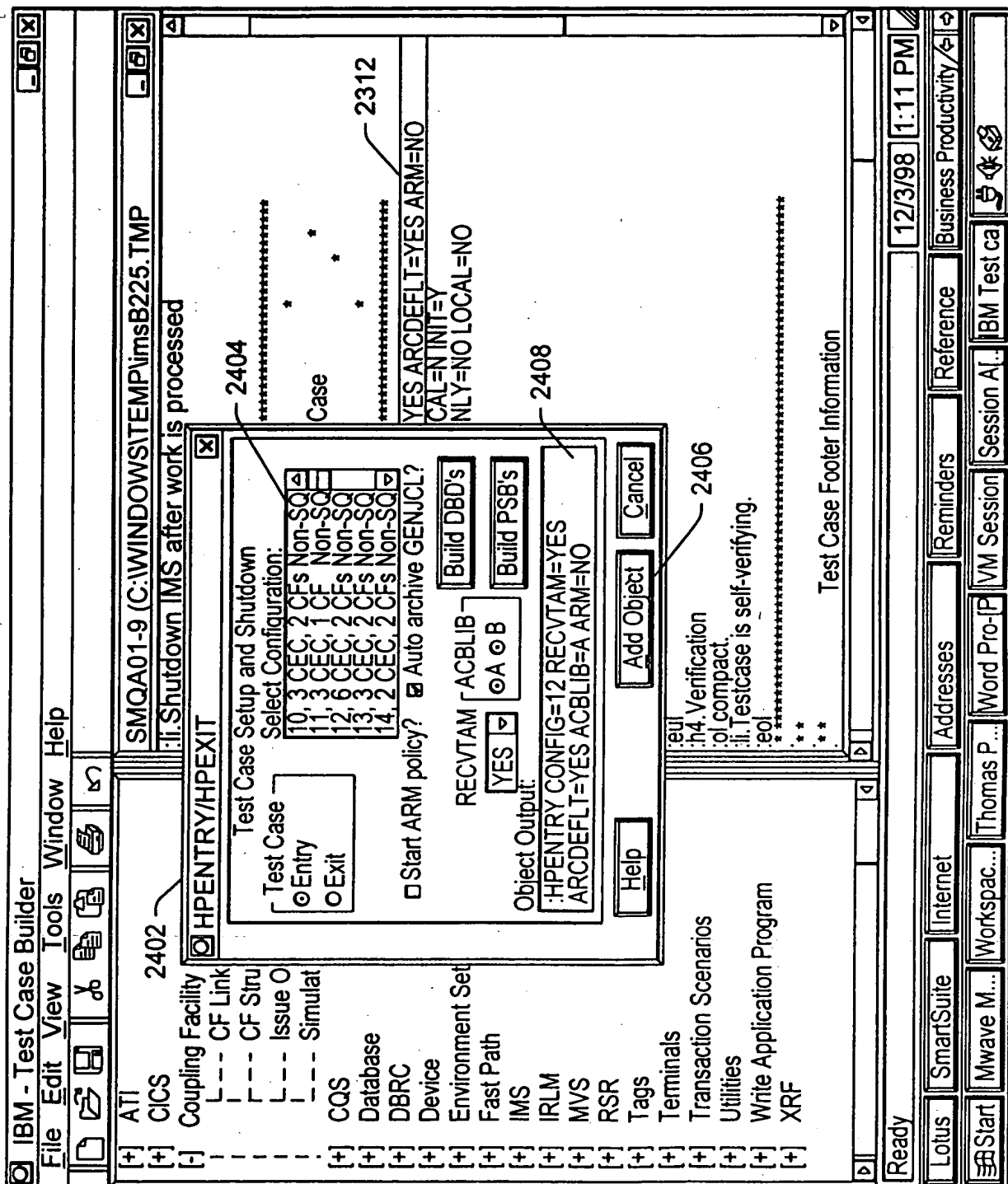


FIG. 24

30/40

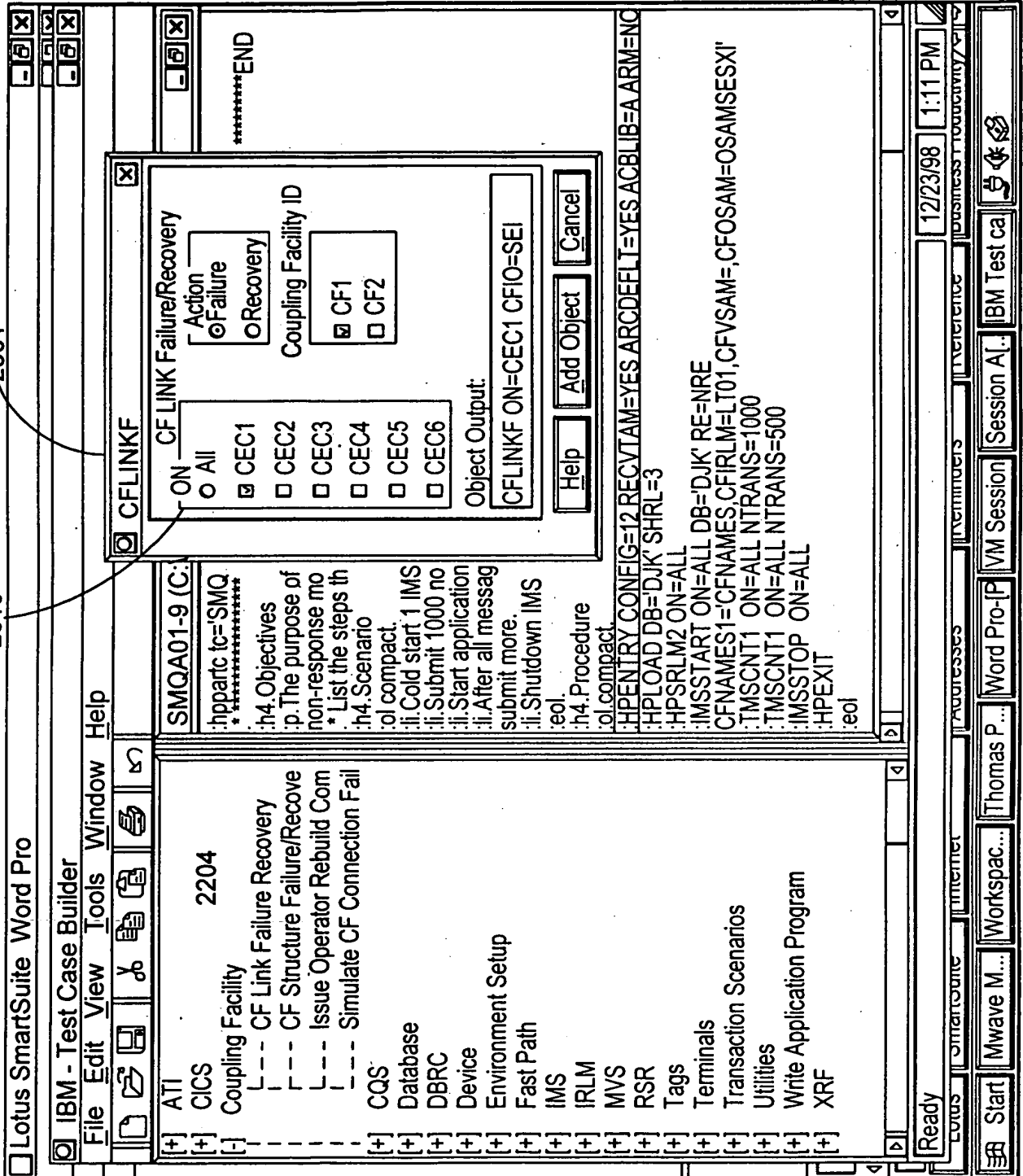
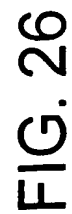


FIG. 25



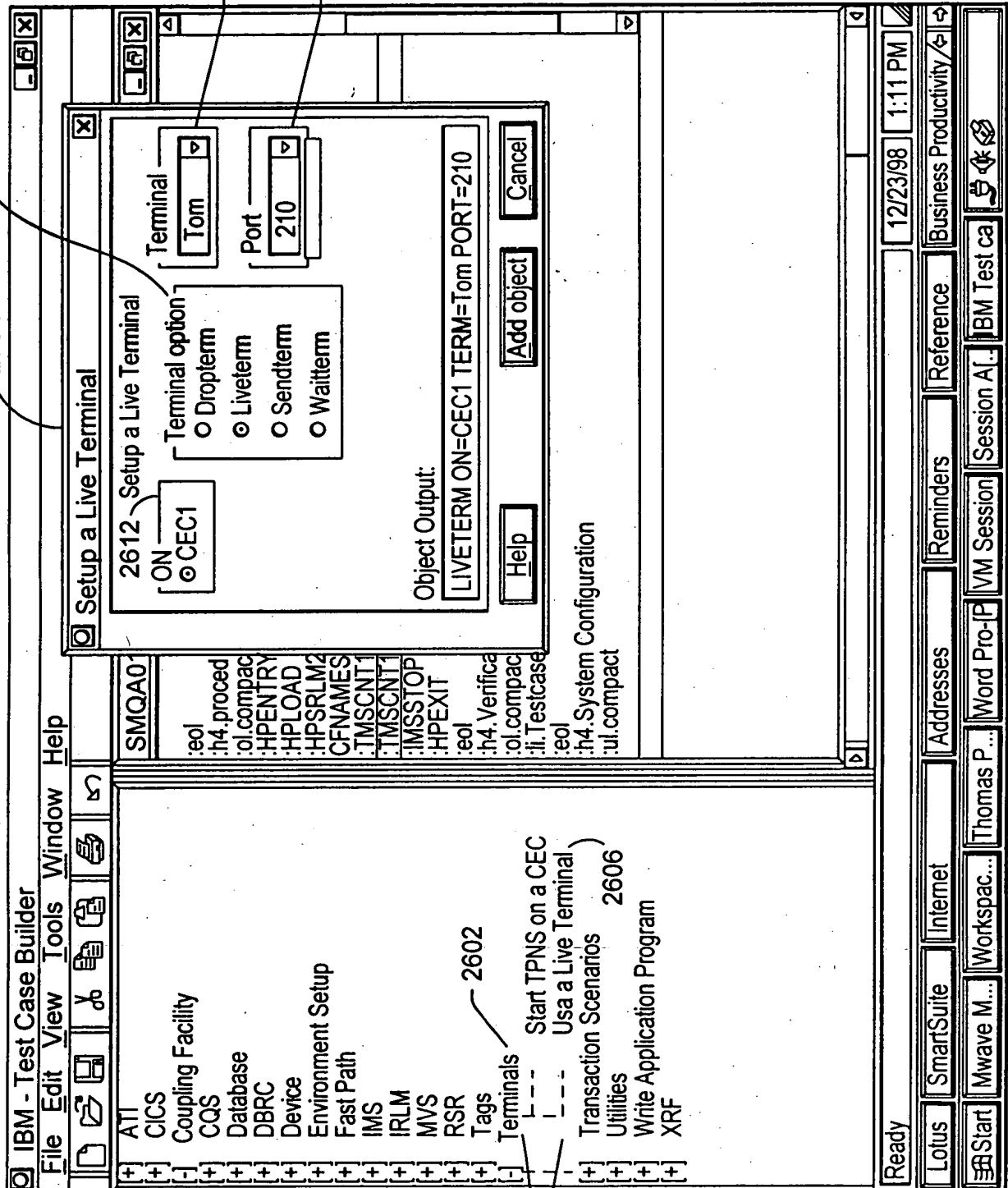
32/40

2608

2702

2614

2616



2604

2602

FIG. 27

33/40

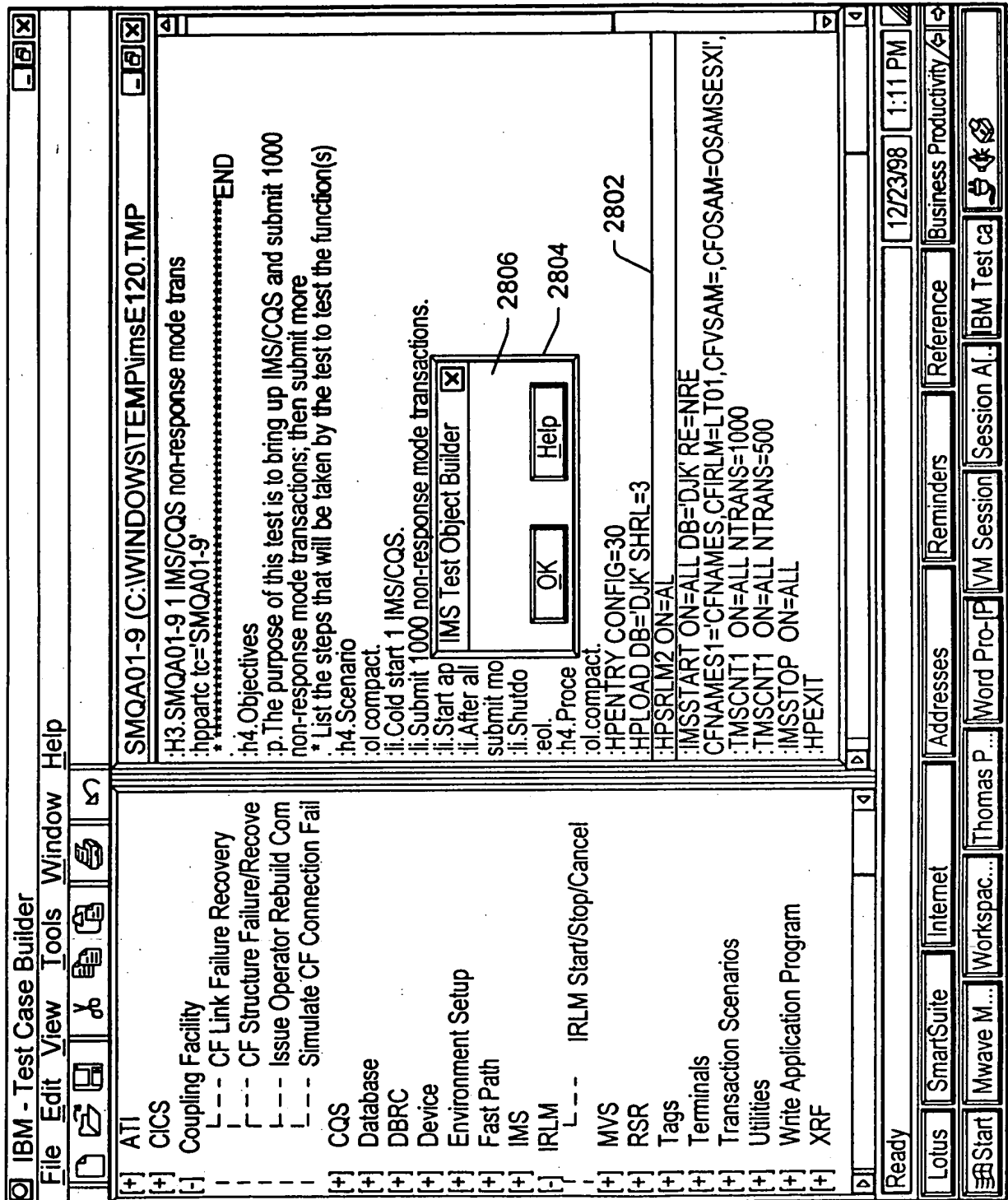


FIG. 28

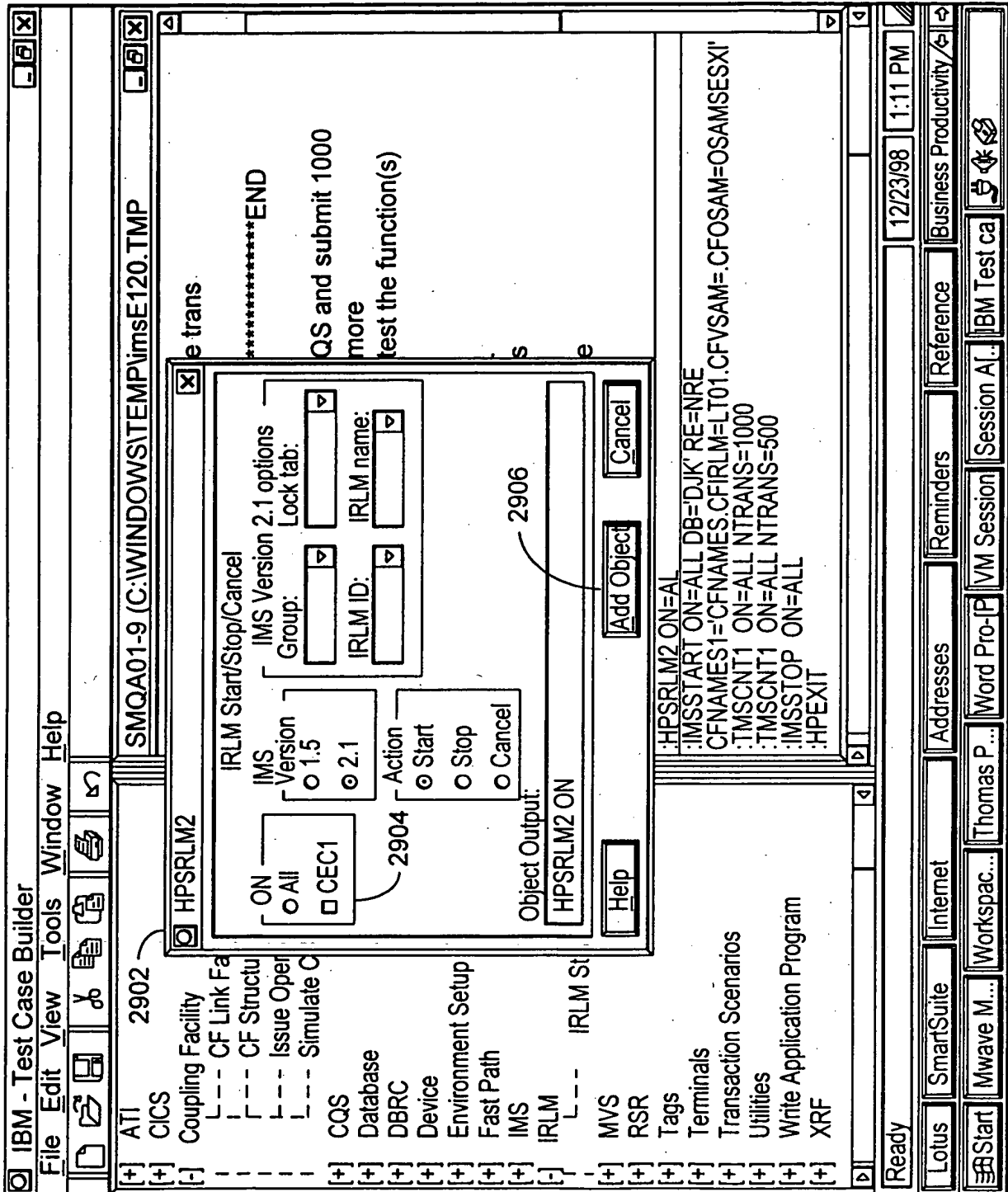


FIG. 29

35/40

```

*****
* IMS Test Object Change History *
* mm/dd/yy - xxx *
*****
*
* *****
* HPSRLM2 Macro Start IRLM 2.1 on indicated CEC *
* *****
.gs rules (vat)
.aa HPSRLM2 HPSRLM2
.dm HPSRLM2 on
.gs attval ON as *onn
.gs attval OPTIONS as *opt
*-----
.if &e'&*onn = 0
.th .go error
.if &u'&*onn = ALL
.th .go all
.if &l'&*onn = 4
.th .go scec
.el .go mcec
*-----
...all
:li.Call Hpcs_Start_IRLMs_21 which will:
:ol compact.
:li.Start IRLM 2.1 on all CECS with a lock structure of LT01
.go cont1
*-----
...scec
.if &e'&*opt = 1
.th .go popt
:li.Call Hpcs_Start_an_IRLM_21 which will:
:ol compact.
:li.Start IRLM 2.1 on &u'&*onn with a lock structure of LT01
.go cont1
*-----
...popt
:li.Call Hpcs_Start_an_IRLM_21 which will:
:ol compact.
:li.Start IRLM 2.1 on &u'&*onn specifying the following options;
&u'&*opt
.go cont1

```

FIG. 30A

36/40

```

*-----
...mcec
:li.Call Hpcs_Start_an_IRLM_21 which will:
:ol compact.
:li.Start IRLM 2.1 on &u'&*onn with a lock structure of LT01
*-----Process Parts Used-----
...cont1
:eol.
.se fn = 'IRLME2N'
.se ft = PROCEDURE
.hpchkpt
.se *fn1 = 'IRLM
.se *fn2 = 2.1
.se fn = &*fn1.&*fn2
.hpadfun
.go done
*-----
...error
:li.+++ERROR in HPSRLM2 INVALID ON Parameter*****
*-----
...done
.dm off
* *****
* end of HPSRLM2 Macro
* *****

```

FIG. 30B

```

HPSRLM2: /*ON=(CECx,ALL) LOCKTAB=1 GROUP=g IRLMID=i */
  Call Parse_variables
  upper on
  if result>1 then return result
  If Total=on1+options1 then return 6
  If on1=0 then return 7
  If options1=0 & on='ALL' then return 18
  If options1=0 & onn>1 then return 18
  data=eighty_blanks
  call Put_line_on_Stack
  Call Put_line_on_stack
  If on='ALL' then do
    Data ='Call Hpcs_Start_IRLMs_21' *****
    Call Put_line_on_stack
  end
  If onn>1 then do
    do j=1 to onn
      work_ec = substr(word(on,j),2,3)
      Call Check_current_ec
      Data ='Call Hpcs_Start_an_IRLM_21' *****
      Call Put_line_on_stack
    end
  end
  If onn=1 & on='ALL' then do
    work_ec = substr(on,2,3)
    Call Check_current_ec
    If options1>0 then do
      Work_String=options
      Call Remove_High_Values
      options = Work_String
      Data = 'OPTIONS='||options||''
      Call Put_line_on_stack
    end
    Data ="Call Hpcs_Start_an_IRLM_21' *****
    Call Put_line_on_stack
  end
  Call Add_Library 'HPC$SUB'
return

```

38/40

[illegible]

```
Hpcs_start_irlms_21:
  Call Save_callers_environment
  Call Hpcs_logit 'Hpcs_start_irlms_21 started'
  If Options==" & Options=="OPTIONS' then do
    Call Hpcs_logit 'Options may not be specified when'
    Call Hpcs_logit 'starting "all" Irlms 2.1'
    goto Hpcs_test_case_aborted
  end
  Irlm_process='Start'
  Call Process_all_irlms
  Call Restore_callers_environment
  Return 0
```

```

/*****&START&*/
/* Routine Name: Process_all_Irlms */
/* Called by: */
/* Parameters passed: */
/* Routines called: */
/* Routine Function: */
/*****&END&****/

```

```

Process_all_lrlms:
    ec=1
    do until forever=true
        CMS 'GLOBALV SELECT MULTIEC STACK EC'ec
        Pull NewEC
        if Index(' 'Sessions' ','NewEC' ')=0 or,
            ec>Maxcec or,
            NewEC="" then do
                Goto Process_all_lrlms_exit
            end
        CMS 'GLOBALV SELECT DOAUTO SET SESSION' NewEC
        CMS 'GLOBALV SELECT DOAUTO SET ECID' NewEC
        Session=NewEC
        Call Process_an_lrlm
        ec=ec+1
    end
end

```

FIG. 32A

39/40

```

      goto Process_all_Irlms_exit
Process_all_Irlms_exit:
      return

```

```

/*****&START&*/
/* Routine Name: Process_an_Irlm */
/* Called by: */
/* Parameters passed: */
/* Routines called: */
/* Routine Function: */
/*****&END&****/

```

```

Process_an_Irlm:
  If Irlm_process='Start' then do
    Call Hpcs_clear_all
    Call Get_irlm_21_Options
    Send 'S IRLME2N,'||Irlm_21_Options
    Wait #1
    Call Hpcs_logit 'Starting IRLME2N on '||NewEC
    Call Hpcs_logit Irlm_21_Options
    hpcs_onerror=onerror
    onerror=False
    CALL DOWAIT '5 1 IRLM INITIALIZATION COMPLETE'
/*   Wait #9:00 Scrhas('RLM INITIALIZATION COMPLETE')
    Wait_rc=rc
    onerror=hpcsonerror
    If Wait_rc=0 then do
      hpcs_onerror=onerror
      onerror=False
      Wait #10 Scrhas('ABEND=S000 U2018')
      Wait_rc=rc
      onerror=hpcs_onerror
      If Wait_rc=1 then do
        Call Hpcs_logit 'Start Irlm issued with active IRLM"s'
        goto Hpcs_test_case_aborted
      end
      Call Hpcs_logit 'Irlm failed to initialize, reason unknown'
      goto Hpcs_test_case_aborted
    end
/*
    Goto Process_an_Irlm_Exit
  end

  If Irlm_process='Cancel' | Irlm_process='Stop' then do
    Call Hpcs_clear_all
    Send 'D A,L'||ENTER

```

FIG. 32B

40/40

```
Wait #10 Scrhas('IEE114I')
hpcs_onerror=onerror
onerror=False
/*****
/* look for RLM after IEE114I message */
*****/
Wait #0 Scrhas('RLM' (HITROW +1:1) (MAXROW : MAXCOL))
Wait_rc=rc
```

FIG. 32C

0955804-091901